Yuanbo Guo (7389051)
COMP5005 — Fall 2014

Homework 3

November 18, 2014

0 Program Description

- Environment: ClojureScript 0.0-2371, OS X 10.10.1
- Source code: https://github.com/gizak/learning-automa/tree/master/assignment3/src/cljs/automaton
- Demo: Add soon

1 Tsetlin

1.1 Simulation

Run Tsetlin machine, settings: $N = 5$, time average cut = 100, instances amount = 1000. The result as following:

Tsetlin Simulation

Run Tsetlin machine, output the simulated p(∞)

c1	c2	N	p(∞)
0.05	0.7	5	[1 0]
0.15	0.7	5	[0.9999000000000001 0.0001]
0.25	0.7	5	[0.9984999999999998 0.0015000000000000002]
0.35	0.7	5	[0.9730999999999999 0.0269]
0.45	0.7	5	[0.8909 0.10909999999999993]
0.55	0.7	5	[0.7080000000000001 0.292]
0.65	0.7	5	[0.5581 0.44189999999999996]

1.2 Min Depth Searching

Using Binary Search Algorithm to find min N ranging from 1 to 100 that makes the accuracy just greater than 0.95. If the result can not be found, then return the nearest value in the range:

Tsetlin Binary Searching Accuracy

Run Binary Search to find min N (range: [1 100]) which makes the accuracy greater than 0.95. If cannot find N, then return the nearest value in the range.

c1	c2	min(N)	accuracy
0.05	0.7	2	0.9949238578680204
0.15	0.7	2	0.9560975609756097
0.25	0.7	3	0.957586169870074
0.35	0.7	5	0.9661414683204828
0.45	0.7	9	0.953172622694726
0.55	0.7	100	0.7999999996916087
0.65	0.7	100	0.5714285714285714

2 Krylov

Comparing Krylov and Tsetlin with several test cases:

Compare Krylov and Tsetlin

Comparing Krylov with (c1 c2) and Tsetlin with (c1/2 c2/2)

c1	c2	N	Tsetlin p(∞)	Krylov p(∞)
0.1	0.3	3	[0.9200000000000002 0.08]	[0.9514999999999999 0.0484999999999995]
0.1	0.5	3	[0.9841 0.0159]	[0.9771 0.0229]
0.4	0.6	4	[0.8922 0.10779999999999999]	[0.8088 0.1912]
0.6	0.7	3	[0.6207999999999998 0.3792000000000001]	[0.5946999999999999 0.4053000000000001]
0.6	0.6	3	[0.5141 0.48590000000000005]	[0.4844 0.5156]
0.3	0.3	4	[0.4671 0.5329]	[0.5172 0.4828]
0.45	0.46	4	[0.48760000000000003 0.5124]	[0.5209999999999999 0.479]

3 L_{RI} Automaton

3.1 Simulation

Run each instance until it is converged ([1 0] or [0 1] state). Here we use $\lambda_R = 0.8$.

LRI Simulation

Simulate LRI machine for given env

c1	c2	mean	p(∞)
0.05	0.7	52.843	[0.932 0.068]
0.15	0.7	59.717	[0.92 0.08]
0.25	0.7	68.742	[0.911 0.089]
0.35	0.7	80.414	[0.887 0.113]
0.45	0.7	98.546	[0.861 0.139]
0.55	0.7	120.068	[0.778 0.222]
0.65	0.7	153.813	[0.619 0.381]

From result, when c1 and c2 are more closer to each other, the mean time is larger and accuracy is lower.

3.2 Search Best λ_R

Use Binary Search to find the best value λ_R (which is also minimal) that reaches 0.95 accuracy. For the sake of speed, we only use 200 simulation instances (which also means the precision is 0.005). Note that this is float number computing, the acceptable accuracy range is [0.95 0.955).

LRI Min λ

Simulate LRI machine for the given env to find λ that makes 0.95 accuracy

c1	c2	λ	acc	mean
0.05	0.7	0.8587386012077332	0.95	77.19
0.15	0.7	0.8665624999999999	0.95	92.235
0.25	0.7	0.940625	0.95	251.18
0.35	0.7	0.89375732421875	0.95	161.75
0.45	0.7	0.9282812500000001	0.95	293.825
0.55	0.7	0.9406233990192413	0.95	538.18
0.65	0.7	0.9812691632658244	0.95	4054.265